Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Original) A method for tracing services on an application server comprising:

identifying a group of services executed on an application server;

for each service in the group, identifying a group of entry and/or exit methods to be traced, the group of entry/exit methods representing entry and exit points to and from service, respectively;

modifying the service's bytecode based on the particular group of entry/exit methods specified;

executing the service; and

registering method invocations of the entry/exit methods.

2. (Currently amended) The method as in claim 1 wherein the application server is a ~~Java 2 Enterprise Edition ("J2EE")~~ server executing object-oriented program code and wherein the group of services comprise ~~J2EE~~ object-oriented services.

3. (Original) The method as in claim 2 wherein one of the group of services comprises a hypertext transport service ("HTTP").

4. (Currently amended) The method as in claim 3 wherein one of the

groups of services comprise a servlet and/or ~~Java~~ Server Page ~~("JSP")~~ service and wherein at least one of the entry/exit methods comprise entry/exit methods linking the servlet and/or ~~JSP~~ server page service to the HTTP service.

5. (Original) The method as in claim I wherein modifying the service's bytecode comprises:

inserting a start method invocation prior to each entry/exit method and inserting an end method invocation following each entry/exit method of the group of entry/exit methods.

6. (Original) The method as in claim 1 wherein registering further comprises: collecting method-related information associated with each of the entry/exit methods responsive to the invocations.

7. (Original) The method as in claim 6 wherein the method-related information comprises a number times that each method of the set of methods is executed.

8. (Original) The method as in claim 6 wherein the method-related information comprises input and/or output parameters associated with each method of the set of methods.

9. (Original) The method as in claim 1 wherein the entry/exit methods

are entry and exit points between a service and an external system.

10. (Currently amended) The method as in claim 9 wherein the service is a ~~Java~~ Connector ~~(JCo)~~ service and the external system is a~~n~~ ~~R/3~~ database system.

11. (Currently amended) The method as in claim 9 wherein the service is an object-oriented ~~J2EE Enterprise Java Bean ("EJB")~~ service and the external system is a non-Web based client.

12. (Original)The method as in claim l wherein the entry/exit methods are entry and exit points between a service and an external database.

13. (Currently amended) The method as in claim 11 wherein the service is a ~~Java~~ Database Connectivity ~~("JDBC")~~ service.

14. (Currently amended) A computer system for tracing program flow of services within an application server , the computer system comprising a memory for storing program code and a processor for processing the program code to perform the operations of ~~comprising~~:

~~a user-configurable plugin module to identify a group of services executed on the application server and, for each service in the group, identify a group of entry~~

~~and/or exit methods to be traced, the group of entry/exit methods representing entry and exit points to and from the service, respectively;~~

~~a bytecode modification module to modify the service's bytecode based on the particular group of entry/exit methods specified; and~~

~~a dispatch unit to register method invocations associated with the entry/exit methods~~

identifying a group of services executed on an application server;

for each service in the group, identifying a group of entry and/or exit methods to be traced, the group of entry/exit methods representing entry and exit points to and from service, respectively;

modifying the service's bytecode based on the particular group of entry/exit methods specified;

executing the service; and

registering method invocations of the entry/exit methods.

15. (Currently amended) The system as in claim 14 wherein the application server is a ~~Java 2 Enterprise Edition ("J2EE")~~ server executing object-oriented program code and wherein the group of services comprise ~~J2EE~~ object-oriented services.

16. (Original) The system as in claim 15 wherein one of the group of services comprises a hypertext transport service ("HTTP").

17. (Currently amended) The system as in claim 16 wherein one of the groups of services comprise a servlet and/or ~~Java~~ Server Page ~~("JSP")~~ service and wherein at least one of the entry/exit methods comprise entry/exit methods linking the servlet and/or ~~JSP~~ server page service to the HTTP service.

18. (Currently amended) The system as in claim 14 wherein~~, to modify the service's bytecode, the bytecode modification module inserts~~ modifying the service's bytecode comprises inserting a start method invocation prior to each entry/exit method and inserts an end method invocation following each entry/exit method of the group of entry/exit methods.

19. (Original) The system as in claim 14 wherein registering further comprises: collecting method-related information associated with each of the entry/exit methods responsive to the invocations.

20. (Original) The system as in claim 19 wherein the method-related information comprises a number times that each method of the set of methods is executed.

21. (Original) The system as in claim 19 wherein the method-related information comprises input and/or output parameters associated with each method of the set of methods.

22. (Original) The system as in claim 19 wherein the entry/exit methods are entry and exit points between a service and an external system.

23. (Currently amended) The system as in claim 22 wherein the service is a ~~Java~~ Connector ~~(JCo)~~ service and the external system is an ~~R/3~~ database system.

24. (Currently amended) The system as in claim 22 wherein the service is an object-oriented ~~J2EE Enterprise Java Bean ("EJB")~~ service and the external system is a non-Web based client.

25. (Original) The system as in claim 14 wherein the entry/exit methods are entry and exit points between a service and an external database.

26. (Currently amended) The system as in claim 24 wherein the service is a ~~Java~~ Database Connectivity ~~("JDBC")~~ service.

27. (Currently amended) The system as in claim 19 ~~further comprising:~~ wherein the program code causes the processor to execute a handler to perform one or more specified output functions on the method invocations and/or the method-related information.

28. (Original) The system as in claim 27 wherein one of the output functions

comprises directing the method invocations and/or method-related information to a display.

29. (Original) An article of manufacture including program code which, when executed by a machine, causes the machine to perform the operations of:

identifying a group of services executed on an application server;

for each service in the group, identifying a group of entry and/or exit methods to be traced, the group of entry/exit methods representing entry and exit points to and from service, respectively;

modifying the service's bytecode based on the particular group of entry/exit methods specified;

executing the service; and

registering method invocations of the entry/exit methods.

30. (Currently amended) The article of manufacture as in claim 29 wherein the application server is a ~~Java 2 Enterprise Edition ("J2EE")~~ server executing object-oriented program code and wherein the group of services comprise ~~J2EE~~ object-oriented services.

31. (Original) The article of manufacture as in claim 30 wherein one of the group of services comprises a hypertext transport service ("HTTP").

32. (Currently amended) The article of manufacture as in claim 31 wherein one of the groups of services comprise a servlet and/or ~~Java~~ Server Page ~~("JSP")~~ service and wherein at least one of the entry/exit methods comprise entry/exit methods linking the servlet and/or ~~JSP~~ server page service to the HTTP service.

33. (Original) The article of manufacture as in claim 29 wherein modifying the service's bytecode comprises:

inserting a start method invocation prior to each entry/exit method and inserting an end method invocation following each entry/exit method of the group of entry/exit methods.

34. (Original) The article of manufacture as in claim 29 wherein registering further comprises:

collecting method-related information associated with each of the entry/exit methods responsive to the invocations.

35. (Original) The article of manufacture as in claim 34 wherein the method-related information comprises a number times that each method of the set of methods is executed.

36. (Original) The article of manufacture as in claim 34 wherein the method-related information comprises input and/or output parameters associated with each

method of the set of methods.

37. (Original) The article of manufacture as in claim 29 wherein the entry/exit methods are entry and exit points between a service and an external system.

38. (Currently amended) The article of manufacture as in claim 37 wherein the service is a ~~Java~~ Connector ~~(JCo)~~ service and the external system is a~~n R/3~~ database system.

39. (Currently amended) The article of manufacture as in claim 37 wherein the service is an object-oriented ~~J2EE Enterprise Java Bean ("EJB")~~ service and the external system is a non-Web based client.

40. (Original) The article of manufacture as in claim 29 wherein the entry/exit methods are entry and exit points between a service and an external database.

41. (Currently amended) The article of manufacture as in claim 40 wherein the service is a ~~Java~~ Database Connectivity ~~("JDBC")~~ service.